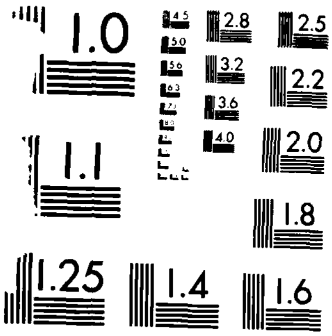
AD-A165 403 CONNECTIONIST MODELS AND PARALLELISM IN HIGH LEVEL VISION(U) ROCHESTER UNIV NY DEPT OF COMPUTER SCIENCE J A FELDMAN JAN 85 TR-146 N00014-82-K-0193 1/1

UNCLASSIFIED F/G 5/8 NL


1.0
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6

(12)

Connectionist Models and Parallelism in High Level Vision

Jerome A. Feldman
Computer Science Department
The University of Rochester
Rochester, NY 14627

TR146
January 1985

Department of Computer Science
University of Rochester
Rochester, New York 14627

12

Connectionist Models and Parallelism in High Level Vision

Jerome A. Feldman
Computer Science Department
The University of Rochester
Rochester, NY 14627

TR146
January, 1985

Abstract

Students of human and machine vision share the belief that massively parallel processing characterizes early vision. For higher levels of visual organization, considerably less is known and there is much less agreement about the best computational view of the processing. This paper lays out a computational framework in which all levels of vision can be naturally carried out in highly parallel fashion.

One key is the representation of all visual information needed for high level processing as discrete parameter values which can be represented by units. Two problems that appear to require sequential attention are described and their solutions within the basically parallel structure are presented. Some simple program results are included.

The preparation of this paper was supported in part by the Defense Advanced Research Projects Agency, monitored by the ONR, under grant No. N00014-82-K-0193, and in part by the National Science Foundation CER grant No. DCR-8320136.

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
146

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
Connectionist Models and Parallelism in High Level Vision

5. TYPE OF REPORT & PERIOD COVERED
technical report

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Jerome A. Feldman

8. CONTRACT OR GRANT NUMBER(s)
N00014-82-K-0193

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Computer Science Department
University of Rochester
Rochester, NY 14627

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Defense Advanced Research Projects Agency
1400 Wilson Blvd.
Arlington, VA 22209

12. REPORT DATE
January 1985

13. NUMBER OF PAGES
33

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)
Office of Naval Research
Information Systems
Arlington, VA 22217

15. SECURITY CLASS. (of this report)
unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Distribution of this document is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES
None.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Students of human and machine vision share the belief that massively parallel processing characterizes early vision. For higher levels of visual organization, considerably less is known and there is much less agreement about the best computational view of the processing. This paper lays out a computational framework in which all levels of vision can be naturally carried out in highly parallel fashion One key is the representation of all visual information needed for high level processing as discrete parameter values which can be represented by units. Two problems that appear to require sequential attention are described and their solutions within the basically parallel structure are included. Some simple program results are also included.

1. Introduction

The human brain is an information processing system, but one that is quite different from conventional computers. The basic computing elements operate in the millisecond range and are about a million times slower than current electronic devices. Since reaction times for a wide range of tasks are a few hundred milliseconds [Posner, 1978], the system must solve hard recognition problems in about a hundred computational time steps. The same time constraints suggest that only simple signals can be sent from one neuron to another. The human information processing system is also adaptable, context-sensitive, error-tolerant, etc., in ways that far outstrip our current computational devices and formalisms.

Students of human and machine vision share the belief that massively parallel processing characterizes early vision. Computational, psychophysical and biological findings agree on the extensive distribution of computation both in spatial organization and along dimensions such as ocularity, size (spatial frequency) and color. For higher levels of visual organization, considerably less is known and there is much less agreement about the best computational view of the processing. But the 100-step argument suggests that, at least for simple tasks, people can do visual recognition tasks much too fast for the processing to be serial. This paper attempts to lay out a computational framework in which all levels of vision can be naturally carried out in highly parallel fashion. In addition to the timing constraint, a biologically plausible model must meet a number of additional computational requirements. The limited number of computational units, their restricted connectivity and very low rate of communication all impose severe constraints which the model attempts to satisfy.

The rest of this introduction informally outlines a proposed model of vision which supports the idea of parallel processing. Section 2 contains a brief review of the connectionist computational model used in the technical sections of the paper. Section 3 describes in some detail the parallel algorithms for high-level visual recognition and how they satisfy a variety of constraints. The final section points out some limitations on parallel processing in vision and lays out a connectionist model of sequential visual attention.

The central problem of vision is taken to be the identification and location of objects in the environment. The critical step in this process is the linking of incoming visual information to stored object descriptions; this is called *indexing* from the analogy of identifying a book from index terms. A system must also identify situations and use this information to guide action. Following the standard usage in computer vision, we divide the problem of visual recognition into three conceptual levels: low, intermediate and high. Low level (or early) vision is characterized by the local nature of its computations. This corresponds to Marr's primal sketch and to anatomical structures from the retina through at least primary visual cortex. Typical low-level operations include image filtering, isolated feature detection and some local relaxation or consistency calcualations. There is no conceptual difficulty in designing massively parallel algorithms for these tasks and several existing systems do various of these tasks in parallel.

Intermediate level vision (ILV) has two major goals: segmentation and the calculation of invariants. The role of ILV is to reduce the incoming visual information to a form that will be effective for the recognition step of high-level vision. One requirement is that the ILV encoding capture the intrinsic properties of objects independent of the particular viewing conditions in the current instance [Barrow & Tenenbaum, 1978]. Recognition also requries that the individual objects in a scene be separated so that they can be individually matched. Segmentation and calculating invariants are mutually interacting computations, depending also on context effects from high-level vision, among other things. Much of the current research in computer vision is concerned with ILV calculations; the development is far too rich to survey here. The computational character of these problems is much more complex than that of early vision; there are unsolved problems in the stability and efficienty of networks for ILV. But the general idea of parallel algorithms for ILV is well understood and a number of partial implementations have been carried out.

The research on parallel algorithms in computer vision has progressed to the point where some general principles are becoming apparent. There appear to be three computational paradigms that are easily adapted to massive parallelism: local calculations, neighborhood function and Hough techniques. Successful applications have been based on combinations of the three computational principles. Calculations that are strictly localized to one area of an image are obviously easy to compute in parallel up to the number of desired results. These include filters and edge detection in early vision and local calculations of, for example, the brightness equation in intermediate level vision.

The second major class of parallel computation is in neighborhood interactions or relaxation [Hummel & Zucker, 1983]. Relaxation in low-level vision has been quite successful, e.g. in smoothing edge and optic flow fields. In a massively parallel system, one can have continuous interactions among strictly local calculations and neighborhood relations. The same idea can be carried over to intermediate level vision as was recognized early by Barrow and Tenenbaum [1978] and is embodied in the smoothness constraint of the MIT school [Poggio & Torre, 1984]. The intrinsic image diagram (Figure 1.1) of Barrow and Tenenbaum continues to be a good characterization of parallel computation in intermediate vision. Notice that the local interactions include not only neighbors in the same plane, but also calculations of other invariants (planes) at the same point in the image.

The third of the comptuational principles is less straightforward but is the key to much of the progress that has been made to date. This principle derives from Hough techniques [Ballard, 1984], which in turn can be traced back to histograms. The idea is simply to count and compare things; the hard part is knowing what to count. Hough techniques and their parallel realizations are particularly good for computing global parameters from plentiful, but noisy local measurements. Typical applications include calculating lines from edge data, illumination angle from surface patches or rigid body motion from local flow vectors. In each case, the answer can be characterized by a small number of parameters: counting the data consistent with each possible answer is an effective solution technique. In a parallel network, all of the relevant counts can be accumulated stimultaneously. The example Hough calculations given above are in early and intermediate vision, but the technique applies to indexing and high level vision as well.

The idea behind applying Hough techniques to recognition is simple. Each visual feature computed in the intermediate stage votes for the objects most consistent with that feature. As an introductory example, we will consider the problem of recognizing a known object that has been transformed and hidden by noise (Figure 1.2). The key constraint here is that the shape of the target is assumed to be known except for possible changes in position, rotation and scale. Under these conditions, the sytem need only solve for four parameters: rotation and scale and translation in x and y. The sequential algorithm for finding the masked object compares evey line in the image with every line in the model. A given image line will match with a fixed model line only for a particular choice of the four parameters. Many choices of parameters get votes, but the (normally unique) choice of parameters that gets a plurality of votes is the correct transform. We will describe the parallel implementation of this scheme in Section 2, after the definitions of our formalism. Of course, the example is greatly oversimplified and the remainder of the paper is concerned with extending parallel recognition techniques to more realistic examples.

2. Connectionist Models

2.1 Background and Overview

Computer science is just beginning to look seriously at parallel computation; it may turn out that conventional programs can be automatically translated into massively parallel networks meeting the hundred-time-step constraint. But no one has yet given the slightest indication of how this might be brought about. An obvious alternative is to start with a computational formalism that has a clear mapping to parallel implementation and attempt to build functional models of intelligent behavior in those terms. If this approach is on the right track, it should prove possible to construct better (clearer, more predictive) models in the parallel formalism than in conventional computer languages. A number of workers in psychology and artificial intelligence are finding these advantages in connectionist models and this paper has definitely required such treatment.

The term "connectionist" comes from a shared assumption of most massively-parallel computational formalisms. This feature arises from the observation that in the psychological quantum of 1/10 second, only a small number (~6) of bits of information can be sent from one neuron to another by spike frequency. This means that the conventional computer mechanism of passing complex symbolic structures cannot be used directly and that the burden of computation must lie on the *connection* structure of the network. There has been a great deal of recent work on the properties of such systems [Amari & Arbib, 1982; Feldman & Ballard, 1982] and on their applications.

There is currently a growing interest in both the abstract properties of connectionist models and in their application to particular problems in the behavioral and brain sciences. The Winter, 1985 issue of *Cognitive Science* is dedicated to this work. To a large extent, applications–oriented efforts such as the current paper use a representation where a single unit represents each item of interest such as a concept, a line segment, and so forth, and are called *localist* models. Another line of work [Hinton & Anderson, 1981] starts from the assumption that concepts are captured by a "pattern of activity" in a large group of units. Most of this work is concerned with general properties of connectionist networks, particularly learning in one form or another [Ackley *et al.*, 1985]. There are a number of positions between the extreme distributed approach and the "grandmother cell" approach to connectionist models. Some of these intermediate representations such as coarse-fine coding play an important role in this paper and are good candidates for describing physiological reality.

2.2 Units and Networks

As part of our effort to lay out a generally useful framework for connectionist theories, we have developed a standard model of the individual unit. These units have a very large number of incoming and outgoing connections and communicate with the rest of the network by transmitting a simple value. A unit transmits the same value to all units to which it is connected. The output value is closely related to the unit's potential and is best described as a level of activation. A unit's potential reflects the amount of activation it has been receiving from other units. All inputs are weighed and combined in a manner specified by the *site functions* and the *potential function* in order to update a unit's potential. A more technical description follows.

A network consists of a large number of units connected to a large number of other units via links. The units are computational entities defined by:

$\{q\}$:a small set of states, (fewer than 10)

p :a continuous value called potential

v :an output value, approximately 10 discrete values

i :a vector of inputs i_1, i_2, ..., i_n (this is elaborated below)

together with functions that define the values of potential, state and output at time $t + 1$, based on the values at time t:

p_{t+1} <----- $P(i_t, p_t, q_t)$

q_{t+1} <----- $Q(i_t, p_t, q_t)$

v_{t+1} <----- $V(i_t, p_t, q_t)$.

A unit need not treat all inputs uniformly. Units receive inputs via links (or connections) and each incoming link has an associated *weight.* A weight may have a *negative* value. A unit weighs each input using the weight on the appropriate link. Furthermore, a unit may have more than one "input site" and incoming links are connected to specific sites. Each site has an associated site-function. These functions carry out local computations based on the input values at the site, and it is the result of this computation that is processed by the functions P, Q and V. The notion of sites is useful in defining interesting unit behavior such as OR-of-AND units where the unit responds to the maximum activation at any of its sites and each site is conjunctive. An example of this is shown in Figure 2.1 where a unit representing a physical size of 4 can be activated by either retinotopic size = 4 AND depth = 1

OR retinotopic size = 2 AND depth = 2. This computational exercise is not, of course, intended as a serious model of size constancy. There are several additional basic computational points that arise from the network of Figure 2.1

Figure 2.1: Relations among depth, retinotopic size, and physical size.

First notice that there is a separate unit dedicated to each possible value of each of the three parameters: depth, retinotopic size and physical size; this unit/value (or place coding) representation is central to all of our models. In this network, as in many others, a consistent state should have only one active value for each parameter. We assume that such networks have mutually inhibitory connections (shown only for depth) among the competing values for each parameter. This mutual inhibition or winner-take-all construction is used in many models and appears frequently in this paper.

Assume for simplicity that the system is viewing a small circle centered on and orthogonal to the line of sight. Then the network of Figure 2.1 specifies a fixed relation among retinotopic size, depth, and physical size. One way to view this is that a given value of depth specifies a *mapping* from retinotopic to physical size; such mappings will be used frequently in the model. The network actually does something computationally much more powerful; it embodies the mutual constraints among the three parameters. If, for example, the physical size of an object (e.g. a person) were known, this would determine the depth value. The computational notion of a network embodying mutual constraints is the fundamental paradigm of connectionist models. The behavior of such systems is characterized by states where a coalition of mutually reinforcing units becomes stable and suppresses its rivals. The two alternative readings of the Necker cube in Figure 2.2 can be nicely interpreted as alternative stable coalitions. Notice that the flip between readings requires the simultaneous reinterpretation of perceptual features at many levels.

Figure 2.2: The Necker Cube

The stable coalition mechanism also has implications for the "grandmother cell" issue. Even the 3-unit loop capturing a size-depth relationship could be viewed as a "pattern of activity" of the three units. More generally, in any connectionist network there will always be many active units forming one or more coalitions. This does not suggest that one can usefully characterize the network in terms of diffuse system states instead of units with particular functions. On the other hand, a unit will participate in several coalitions and need not have a simple response pattern. There are both biological and computational advantages to using the simultaneous activity of multiple units to code some information of interest. Notice also that a coalition is not a particular anatomical structure, but a temporarily mutually reinforcing set of units, in the spirit of Hebb's cell assemblies [Jusczyk, 1980].

Another use of these networks is in the parallel realization of Hough transform techniques [Ballard & Brown, 1982]. Figure 2.3a depicts the well known scheme whereby edges (on the left) each "vote for" the slope (θ) and distance (ρ) of the line most consistant with the edge. Figure 2.3b shows how a simple network can carry out these calculations in a fully parallel fashion. The networks for the generalized Hough techniques used in Figure 1.2 are more complex, but follow the same principles

[Ballard *et al.*, 1984]. The idea of computing the best fit in a discrete parameter space is central to this paper and appears to be a key to parallelism in intermediate and high level vision.

The 100 billion units that comprise the human brain also impose constraints on our models. For example, suppose we wanted to represent 10 values each of ten low-level visual features such as position, orientation, hue, contrast, motion, etc. Having a separate unit for each vector of values would require 10^{10} units which is clearly too many. Suppose instead we had units which were precise in only one dimension. Then we would need only 10 x 10 units but it would take the simultaneous activity of ten units to specify a full vector of values. There are a range of intermediate constructions [Hinton, 1981; Feldman & Ballard, 1982]. One of these techniques (coarse-fine coding) appears close to the coding used in primary visual cortex, where units are broadly tuned in several dimensions and fine-tuned in one stimulus dimension.

2.3 Memory and Change

In the previous section, we saw how fixed connectionist networks could be designed to compute functions and relations quite efficiently. These fixed networks could have a certain amount of built-in flexibility by explicitly incorporating *parameters*. One can view the depth networks of Figure 2.1 as computing the physical size of objects from the retinotopic size, parametrized by depth.

But there are also a number of situations where it does not seem plausible to assume the existence of either fixed or parametrized links. Obvious, though artificial, examples are the paired-associate tasks with nonsense syllables used by psychologists. A closely related real task is learning someone's name or the Hebrew word for apple. One cannot assume that all the required connections are pre-established, and it is known that they do not grow rapidly enough [Cotman *et al.*, 1981]. What does seem plausible is that there is a built-in network, something like a telephone switching network, which can be configured to capture the required link between two units. We refer to this as establishing a "dynamic connection" in the uniform network. We are assuming (as is commonly done) that the weight of synaptic connections cannot change rapidly enough to do this, so that all dynamic connections are based on changes in the potential (p) and state (q) of individual units. The other basic constraints that we impose on possible solutions are that units broadcast their outputs and that there is no central controller available to set up the dynamic connections. These assumptions differ from those in the switching literature, and the results there don't carry over in any obvious way. The assumption is that only one dynamic connection is made at a time, but that several (e.g. 7 ± 2) must be sustainable without cross talk.

A sample task is to make arbitrary dynamic connections between two sets of units labelled A. . .Z and a. . .z respectively. These could be words in different languages, paired associates, words and images, and so on. Figure 2.3 depicts the situation for three units on each side.

The problem is how to establish, for example, the link B-c without also linking, e.g. B-b, since the network is originally uniform. More precisely, we require an algorithm which, given the simultaneous activation of B and c, will establish p and q values in the units of our network such that (for some time) activating B will stimulate c but not a or b. For the most part we have considered symmetric networks where the "dynamic connection" B-c will also have the activation of c stimulate B and not A or C. It should be clear that primitive units without any internal state (memory) will not be usable in such tasks.

The basic solution to the dynamic link problem in connectionist networks relies upon mutual inhibition between the alternative inter-units. For notational convenience, we represent this situation as an array of units, with the understanding that the array is a winner-take-all network. If the only active link were B-c, then only the three starred units would be active. Uniform dynamic link networks like those described above are an essential part of our model of how visual objects are linked to locations.

Figure 2.3: Uniform dynamic link network.

The network of Figure 2.3 uses a separate intermediate unit dedicated to each possible pairing. The starred unit for B-c is in two winner-take-all networks, the column which is "inputs to c", and the "outputs from B" net which is drawn in explicitly. When B-c is active, it blocks all other uses of both B and c, which is the desired effect. The fact that our solution requires N^2 intermediate nodes to connect 2N units makes it impractical for linking up sets of 10^5 units like an educated person's vocabulary. There are, however, more complex interconnection networks which require about $4N^{3/2}$ units [Feldman, 1982]. That paper also gives detailed descriptions of the unit computations required and some examples.

2.4 Random Interconnection Networks

There are both anatomical [Buser, 1978] and computational reasons for looking carefully at random interconnection schemes. One possibility is to use random interconnection networks (in place of the uniform networks above) to dynamically connect arbitrary pairs of units from two distinct layers. As before, each unit is postulated to have links to some large number of intermediate units, whose role is strictly a linking one. In any random connection scheme there will be some finite probability that the required path is simply not present. The remarkable fact is that this failure probability can be made vanishingly small for networks of quite moderate size [Feldman, 1982]. The idea is to have k (two or more) layers of intermediate units so that there is a tree of B^{k+1} links across the network, where B is the outgoing number of branches from each unit. This result has been known for some time and has been used as the basis of a proposed highly parallel computer [Fahlman, 1980].

It is premature to speculate on the degree to which the association cortices of animals are more like the uniform or random networks (if either) but we can say something about the computational advantages of each. Uniform networks appear to be most useful for maintaining many simultaneous dynamic links which are easily turned on and off. They could only be expected to occur in well-structured stable domains because of the strong consistency requirements. In general, we would like to view uniform dynamic links as a mechanism roughly equivalent to modifiable or conjunctive connections where the number of possibilities is too great to wire up directly.

Random interconnection networks are not as stable and predictable as uniform ones, but have some other advantages. The lower requirements on the number and precision of wiring of intermediate units are clearly important. But the most interesting property of the random networks is the relative ease with which they could be made permanent. Suppose thaı instead of rapid change we wanted relatively long term linkage of units from the two layers. Our model specifies that this must be done by changing connection weights w_j. The point to be made here is that the random networks already have some units biased towards linking any particular pair from the two layers. By selectively strenghthening the active inputs (on command) of the most appropriate units, the network can relatively quickly forge a reliable link between the pair. The details of how we propose that this comes about are given in Feldman [1982]. Of course, once this has happened, the network will not be able to represent competing dynamic links, but its ability to capture new pairings will remain intact until a large fraction of the nodes are used up (cf. [Fahlman, 1980]).

The fact that random (as opposed to uniform) interconnection networks could be readily specialized suggests that random networks may play an important role in permanent change and memory. After enough training, the originally random interconnection network would become one in which there was essentially a hard-wired connection between particular pairs of units from the two spaces.

The problem with this scheme as a proto-model of long term memory is that most of our knowledge is structured much more richly than paired associates. It is technically true that one can reduce any relational structure to one involving only pairings, and Fahlman [1980] suggests that the best current hardware approach is along these lines. But the intuitive, psychological and physiological [Wickelgren, 1979] notions of conceptual structures involve the direct use of more complex connection patterns. It turns out that the results on random interconnection layers extend nicely to the more general case.

The proposed solution to recruiting units to capture new associations depends on the properties of randomly connected graph structures. It turns out that a random graph of N nodes each connected to about $\sqrt{N}$ others has very useful statistical properties. (Think of N as about 1,000,000 and $\sqrt{N}$ as 1,000 for neural networks). If some small number of nodes (say 30) are chosen at random, the important question is the probability of there being a small network that includes the chosen nodes and is sufficiently well connected to form a *stable coalition* (as defined in Section 2.2). If there is such a sub-network, it could be recruited to represent the new concept whose features are represented by the originally chosen nodes. For random networks of the type described above, the probability of there being a binding sub-network is quite high and the dynamics of recruiting the concept structure also appear to be feasible [Feldman, 1982]. Notice that the concept would be represented by a few dozen units, providing another example intermediate between unit/value and diffuse representations.

This is the basic mechanism that we believe supports associative learning and appears to be close to what Wickelgren [1979] had in mind. If random chunking networks can be made to support short-term associations through coalitions, the usual weight-changing algorithms would enable the associations to be made permanent [Sutton, 1981; Feldman, 1982]. Such mechanisms are postulated to underlie the grouping of an object instance with its properties and the structuring of a complex scene into a "situation" network. More generally, the technical notions of uniform and random dynamic links are essential to all local connectionist models, the current model of vision and space being the most comprehensive effort to date.

3. Parallel Visual Recognition

The central problem addressed by this paper is how a visual system can recognize objects and situations with a delay of less than 100 times the speed of its basic computing units. The technical mechanisms presented in the previous section will enable us to look at this problem in more detail. We have already seen how a system built along the lines suggested in Section 2 could be made to recognize a fixed, two-dimensional object in a very cluttered scene. Many of the same ideas will carry over to more complex vision problems, but there are also a number of new techniques needed.

One major addition to the notions examined above is the introduction of hierarchical object descriptions. Outline figures can be described directly in terms of their component lines, but this becomes infeasible in more realistic visual enevironments. Daniel Sabbah [Sabbah, 1985] has developed a connectionist system that demonstrates how the concepts of Section 2 can be used to recognize objects in a fairly complex domain, that of Origami objects. Origami-world was introduced by Kanade [1978] and shown to be an interesting task domain, especially for studying the role of skewed symmetries for determining the orientation of planes in space. Sabbah's work explores the use of connectionist networks to build fast and reliable solutions to this problem.

Figure 3.1 shows the behavior of the program when given a line drawing depiction of an Origami chair. The crossed lines are an indication of where the program has deduced the presence of a plane in space. Figure 3.2 presents a hierarchical description scheme used in the system. The need for a hierarchical description should be clear; a single L-joint in the image could correspond to a huge range of appearance possibilities for the chair. The program includes intermediate level networks that compute more complex joints and ones that compute parallelograms in the image. These features can then be combined to provide effective indexing for objects like the Origami chair. Sabbah's program actually does rather more than this. It incorporates "top-down" links from a 2-dimensional shape to the L-joints that could give rise to that shape (in a fixed position). This enables the system to be somewhat noise resistant and helps deal with the problems of occlusion. The program also uses T-joints as explicit occlusion cues. This enables it to deal correctly with scenes containing a modest amount of occlusion. The treatment of more general scenes with occlusion is one of the unsolved problems discussed in Section 4.

Sabbah's Origami world system used hierarchical descriptions and three dimensions, but in one way was less sophisticated than Ballard's object finder. The

Origami program did not explictly compute the viewing transform and actually had to incorporate separate networks for the different appearance possibilities of objects like the Origami chair. We have been working recently on a more general connectionist vision system design that is conceptually adequate to deal with a significant range of natural scenes. This is much more complex and is not completely implementable, but a description of it should help provide insight into the problem and our proposed solutions. The proposed parallel vision model and its relation to behavioral and biological findings have been described in detail elsewhere [Feldman, 1985]. For our purposes, all that is required is an overview of how the system functionality is divided among four representational frames (Figure 3.3).

The representation of information in the first frame is intended to model the view of the world that changes with each eye movement. The second frame must deal with the phenomena surrounding what has been called "the illusion of a stable visual world." A static observer has the experience of (and can perform as if he held) a much more uniform visual scene than the first foveal-periphery frame is processing at each fixation. One can think of the second frame as associated with the position of the observer's head; this is an oversimplification, but conveys the right kind of relation between the two frames. Of course, neither of these frames is like a photographic image of the world. Light striking the retina is already transformed, and the layers of the retina, the thalamus, and the visual cortex all compute complex functions. The crucial difference between these two frames is that the first one is totally updated with each saccade and the second is not. The current model also assumes that the first (*retinotopic*) *frame* computes proximal stimulus features and the second captures distal (constancy, intrinsic) features as well as being stable; it is therefore called the *stable feature frame.*

The third and fourth representational frames are both multi-modal and thus unlikely to be the same as the first two. The third representation is not primarily geometrical and will be described in the next paragraph. The fourth, or *environmental frame*, is intended to model an animal's representation of the space around it at a given moment. It captures the information that enables one to locate quickly the source of a stimulus from sound, wind, smell, or verbal cue, as well as maintaining the relative location of visual phenomena not currently in view. For a variety of reasons, the model proposes a single allocentric environmental frame which gets mapped, by *situation links*, to the current situation and the observer's place in it.

The final representational frame to be considered is the observer's general knowledge of the world, including items not dealing with either vision or space. We follow the conventional wisdom in assuming that this knowledge is captured in propositional (relational) form, modeled in our case by a kind of semantic network. One class of knowledge encoded will be the visual appearance of objects encoded as a collection of relationships among primitive parts. These descriptions have much of the character of Minsky's conceptual frames [1975] and of the object-centered frames of, for example, Ballard [1984] and Hinton [1981a]. Since the other three representations are geometrically organized, we will refer to the collection of semantic knowledge as the *world knowledge formulary*, to emphasize its nature as a collection of conceptual relations. The formulary will carry much of the burden for integrating information from the other three frames and is far from adequately worked out in this paper. But all we need for now is the notion that the network representation is likely to be quite different from that of the retinotopic, feature, or environmental frame. All of this indicates that even a provisional model of vision and space will require at least four representational frames.

The central problem is linking visual feature information with the knowledge of how objects in the world can appear. The problem of going from a set of visual features to the description of a situation will be called the *indexing problem*, by analogy with looking up something in an index. The small world we will consider in detail has exactly six distinct visual features each with 10 possible values. Assume for now that any object in the small world can be characterized by some particular set of values for the six features. This would mean that each object has a distinct 6-digit visual code (not unlike a zip code). If the system could always reliably extract the values for the visual features, it would not be hard to identify which objects were in which places in the current environment. No additional problems would arise if some objects had multiple codes among the 10^6 = 1,000,000 available. But the system, as specified, would totally break down if two objects needed to share the same code, i.e. looked identical relative to our set of features and values. We will have to address the question of ambiguous feature sets later.

The six particular visual features which we have chosen are intended to elucidate the major scientific problems in intermediate level vision and would not be the best choice for a practical computer vision system. We assume for now that the best value at each position of the current view is continuously maintained by parameter network computations [Ballard, 1984] which will be elaborated below. Some of the parametrizations are turning out to be rather subtle. For example, it appears that natural textures can be well characterized by fractal parameters [Pentland, 1984]. Features such as size and shape, which cover several units, are assumed to be represented by a single unit at the center of the region covered. Of course, the problem of breaking up the feature space into meaningful regions is a central one and the model will have to address it in detail.

The six visual features used in indexing are the following: lightness, hue, texture, shape, motion, and size. Obviously enough, ten values of these features (even in logarithmic scales) is not enough to characterize visual appearance in the real world; but the small world is rich enough to exhibit most of the required problems. The model assumes that the six features are continously represented in six parallel 10 x 10 arrays which are intended to map the currently visible external world. There is also assumed to be a (10 valued logarthmic) depth map maintained as part of the same structure (Figure 3.3). The depth map is needed for calculating constancy features such as object size and is also used directly in mapping the environment. The depth map is assumed to be calculated cooperatively with the six feature planes, using binocular and other cues. These seven parallel arrays, along with some auxiliary structure, comprise the *stable feature frame* which is one of the four cornerstones of the model.

Our first notion of appearance models was that each object could be characterized by one or more sets of feature values. For objects that are sufficiently simple, this is not a bad approximation. You can probably name an object that is an approximately 1.5" white sphere and uniformly pock-marked even before seeing it hook into the rough. But for complex objects like a horse or Harvard Square, the single feature set isn't even the right kind of visual information. Our way of handling the appearance models for complex objects and situations is taken from current AI practice. We assume that the appearance of a complex object is represented (as part of one's world knowledge) as a network of nodes representing the "appearance possibilities" of simpler components and relationships among them (Figure 3.5). There are several unsolved technical questions about the number of separate views maintained, and how much flexibility should be encoded in a description, but the general idea of hierarchical network is all we need at the moment.

Recall that the naive version of indexing was to use the 6-digit visual feature code to look up the name of the object with that description. Complex objects are assumed to be composed of parts, each part being either another complex object or a *visual element* that can be indexed by the 6-digit code. Now recall that all of our structures are assumed to be parallel and continuously active. This means that "indexing" can be continuously in progress between different areas of the feature frame and networks of visual appearance knowledge in the world knowledge formulary. The crude version of this idea is to assume that each set of visual features (for a point in the 10 x 10 feature frame map) picks out (indexes) the visual element which is appropriate. If this were to happen, it is not hard to see that a complex visible object would have many of its visual elements selected simultaneously and should therefore be recognizable. Recognition of an object or situation is modeled as a mutually reinforcing coalition of active nodes in the world knowledge frame. The mutual excitation of feature and model networks also involves top-down, *context*, links from visual elements to the feature units that are appropriate.

In order to make these notions more precise and eliminate the ghosts from our machine, we must describe all of this in considerably more detail, using the technical definitions of Section 2. The various components of both the feature frame and world knowledge frame will be elaborated in terms of the "units" of Section 2. Obviously enough, we will need separate units for each of the 100 spatial positions in each of the seven separate maps. In fact, it is also important to follow the unit/value principle and require a separate unit for each value of each cell in the maps above, giving a total of 7000 units. Following the connectionist dogma, we assume that visual elements are units which are connected to the appropriate set of visual-feature-value units. For example, Figure 3.4 shows how golf and ping pong ball descriptions in the world knowledge frame might be connected (indexed) by visual features. Having a separate unit for each feature value and for each visual element allows the system to simultaneously maintain several possible interpretations of the scene. This essential information processing requirement is unachievable in conventional computational schemes.

It is easy to see how to make connections do the same job as the index codes. Each code for a visual element is mapped into a conjunction of links from units representing the appropriate value of each feature. A visual element with multiple codes has several disjunctive "dendrites," one for each code. Visual elements that are part of a complex object are also linked into a network for representing the appearance of the object (Figure 3.5).

Complex objects (and situations) are represented as networks (in the world knowledge formulary) of nodes describing visual elements or other complex objects. There are tremendous problems of several different kinds in these semantic network models; these are discussed by Shastri & Feldman [1984]. Our goal here is just to provide a plausible (though crude) model of how network representation of visual appearance could fit in the four-frames paradigm.

The basic idea is that each visual element of a complex object is represented by a node that corresponds to a particular set of feature values as computed in the feature frame. Since indexing from features to elements occurs in parallel, there will usually be several simultaneously active element nodes for a complex object currently in view. This simultaneous activation of subparts will tend to cause the correct complex objects to be activated, independent of the details of how the relationships among the subparts are modelled. When we consider the details of complex object representations, a number of difficult technical problems arise. This is discussed in detail in Hrechanyk & Ballard [1982], and we will be content here with a loose discussion, based on the example of representing the visual appearance of horses. Recall that the world knowledge formulary visual appearance models are far from complete -- they are more like a verbal description of something not currently in view.

Obviously enough, the side and bottom views of a horse have relatively little in common. Even within the side view, the horse could appear in a variety of orientations and scale configurations and the relative positions of its subparts could also differ considerably. We must also account for the fact that there could be several distinguishable horses in a scene and that some of these may be partially occluded. Our current solution, depicted in Figure 3.5, involves instance nodes, separate subnetworks for different views and cross-referenced structural descriptions. The prototype horse has a general hierarchical description where, e.g., the trunk is composed of a body, legs and a tail. What visual elements might be involved in recognizing a horse will depend on whether it is a front, side or other view. Thus the matching process would select together a prototype and a view which best matched the active visual elements. As always, there is assumed to be mutual inhibition among competing object descriptions and view nodes.

4. Limits on Parallelism

The main issue addressed in this paper is how much of all this could be done in parallel with reasonable amounts of hardware (of the scale of the brain). There appear to be two separate places in the system where parallelism breaks down. The first problem that may require sequential processing is the abstraction of visual features from their spatial location in the intrinsic image or stable feature frame. The basic problem is combinatorial. Suppose one wanted to use collections of feature values to index into world knowledge. Even with as few as six features having ten values each, one gets 10^6 separate primitives. If we had a separate unit for each point in a 1000^2 image (typical of the retina or moderate resolution images), it would take 10^{12} units, which is more than our limit for the entire system. Realistic numbers for features and values clearly preclude this computational solution.

The fact that there are not enough units to denote one to each collection of features at each point in the visual field is a classic problem for parallel models. One common form of the question is to ask how a system can avoid detecting a red square when a red circle and a blue square are simultaneously present [Feldman & Ballard, 1982]. Our solution to this problem involves conjunctive connections (Section 2.2) and spatial coherence. The idea is to employ spatially invariant units to detect collection (here pairs) of specific feature values e.g. a pocked sphere. There is assumed to be only one "pocked sphere" unit, but the activity of its inputs is conjoined so that two active inputs must be from the same point in space. The number of units required for this coding is rather modest. For the small world of six ten-valued features, there would be 15 x 10^2 or 1500 feature-pair units as opposed to the 100,000,000 that would be needed to encode a feature vector at every position in the 10x10 field. There are a variety of other ways to reduce the number of required units to a feasible number, but they all share the problem of vulnerability to confusion (crosstalk).

One abstract way to envision the crosstalk problem is to notice that any encoding of the space of feature vectors causes some sharing of codes. The system will not be able to distinguish two inputs that map to the same code. As a concrete example, the simultaneous apppearance of an orange and a flying ping-pong ball might activate golf-ball in the network of Figure 3.4. People do form such illusory conjunctions under certain conditions [Treisman, 1982], but the problem does not arise in normal vision. This is partly explainable by mutual inhibition by other percepts, but there are good reasons to believe that sequential processing is used to avoid crosstalk. If the system could restrict input so that it came from only a small area of the field, the problem of potential crosstalk would be greatly reduced. This idea of sequentially focussing attention appears to be *universally applicable* to connectionist networks and fits quite nicely with the psychologist's notion of cc ert attention [Posner & Cohen, 1984]. There are a number of open issues [Feldman, 1985], but it does seem that sequential attention is the best known solution to the problem of crosstalk in a parallel system of bounded size.

The other place where parallelism in higher level vision appears to break down is related, but more subtle. Consider the recognition network for horses in Figure 3.5. Also assume that the visual feature sets are represented independently of their position, as shown in that figure. A network like Figure 3.5 would respond positively to an image in which the features of a horse were all present, but were totally scrambled in position and relative orientation, because the features have been abstracted from their spatial location. The relational information among features has been lost in the parallel indexing process. This problem of illusory conjunctions and misguided recognition does arise in special situations [Treisman and Gelade, 1980; Thompson 1980] but not in normal vision.

There is one solution to this scrambled image problem that will occur immediately to any vision researcher -- junction features. One could add to our mechanisms recognizers for junctions of features analogous to the L- and T- joints of blocks world vision. An image would have to match not only the individual features but also the junctions to be recognized. This does help considerably, but some confusions still can arise, particularly in scenes with occlusion. The only general solution we have found to this problem again requires sequential processing.

Even with sequential processing allowed, verifying the structural relations constituting something like a horse is not a trivial problem in connectionist modelling. A program recently developed by D. Plaut [1984] points out some of the difficulties in this task and how they may be be overcome. Because our simulator was so slow, the simulation is carried out in a pico world where visual input is confined to a hexagonal grid of ten cells (Figure 4.1). Figure 4.1a depicts a toy train that is composed of a large and a small shiny red cylinder and two small dull brown spheres. The individual visual features are idealizations and their computation was not part of the program. Figure 4.1b shows how the individual features are combined (in parallel) to form feature collections which, in turn, index possible models of simple toys. A technical point is that the program employs feature-pair units (such as sml. sph. for small sphere) that force the two feature values to come from the same spatial position for the unit to become active. In such a network a small brown dull sphere anywhere in the image will cause the appropriate unit to be activated. As we mentioned before, if enough of the parts of the toy train are activated, the "train" node will be effectively indexed. But this is not the end of the recognition process.

For one thing, the train in Figure 4.1 is translated and rotated. As we discussed earlier, there are good ways to compute image to model transforms in connectionist networks; Figure 4.2 depicts such a network for our pico world. The idea here is to use the position and orientation of some major part of the object, here the train body, to determine the transform. By focussing attention on the large red shiny cylinder and exploiting the top-down connections back to spatial location units, the system can determine (i.e. activate) the parameters of the viewing transform. This obviously must be done sequentially for each object in the scene, but is only preliminary to the main process of model verification.

Recall that the central problem was verifying the relationships among the component parts of an object such as a toy train. Notice that junctions alone will not distinguish the toy train from an object with the smokestack and one of the wheels switched. Our solution to the structure mapping problem involves sequentially verifying that each part is in the appropriate relation to its neighbors. The connectionst implementation of this scheme is suggested by Figure 4.3.

The verification process begins with the principal part used to compute the viewing transform as in [Marr & Nishihara, 1978; Hrechanyk & Ballard, 1983]. Each subpart is checked in turn, using a connectionist routine [Shastri & Feldman, 1984] to select a particular part, compute where it should be and then focus attention on the appropriate part of space. For each part, its relative position in the model and the current viewing transform combine to determine its expected position in the image. In Figure 4.4, the train top is one unit to the upper right of the body in the model, and the viewing transform is a rotation of 60° and a translation of (2,2). The network shown combines these values to activate the position (3,1) as the expected location of the top. By allowing input only from this location (i.e. attending only to it), the program is able to test for the presence of the right primitive in the right place. The details of this process and its extension to more complex problems is discussed in [Hrechanyk & Ballard, 1983; Plaut, 1984].

Obviously enough, the current program is very primitive, but it does show a number of things. Parallel indexing from intrinsic features appears to be feasible and extensible to large problems. The parallel computations of a viewing transform from model to image is feasible at least for unoccluded objects with an identifiable orientation. Relational information lost in the parallel indexing process can be regained by sequentially attending to parts of a figure and the rest of the mechanism continues to be fully parallel. The major open questions involve extending these ideas to realistic domains, including occlusion and relating these algorithms to the intriguingly similar results on human eye-movements and attention.

Although the application of connectionist models to vision is at a very early stage, the results have been quite encouraging. It appears that low, intermediate and high level visual processing can all be expressed well in the formalism, often better than in any other known way. The computational limitations that seem to be inherent and the natural solution to them map nicely onto what is known from brain and behavioral studies. Our current efforts include extending the ideas to harder problems, carrying out more detailed simulations (using a parallel computer) and working closely with colleagues in other disciplines to test specific hypotheses on natural vision.

References

Ackley, D.H., G.E. Hinton and T.J. Sejnowski, "A learning algorithm for Boltzmann machines," to appear *Cognitive Science*, Winter, 1985.

Amari, S. and M.A. Arbib (eds.), *Competition and Cooperation in Neural Nets*, Vol. 45 of *Lecture Notes in Biomathematics*, S. Levin (ed.). Berlin: Springer-Verlag, 1982.

Ballard, D.H., "Parameter networks," *Artificial Intelligence*, 22, 1984, 235-267.

Ballard, D.H. and C.M. Brown. *Computer Vision*. Prentice Hall, 1982.

Ballard, D.H., G.E. Hinton and T.J. Sejnowski, "Parallel visual computation," *Nature*, vol. 306, no. 5938, 3 November 1983, 21-26.

Ballard, D.H., A. Bandyopadhyay, J. Sullins and H. Tanaka, "A connectionist polyhedral model of extrapersonal space," *Proceedings*, IEEE Workshop on Computer Vision: Representation and Control, 18-24, Annapolis, MD., 1984.

Barrow, H.G. and J.M. Tenenbaum, "Recovering intrinsic scene characteristics from images." In Hanson, A.R. and E.M. Riseman (eds.), *Computer Vision Systems*. NY: Academic Press, 1978.

Brown, C.M., "Computer vision and natural constraints," *Science*, vol. 24, no. 4655, 22 June 1984, 1299-1305.

Feldman, J.A., "Four frames suffice: a provisional model of vision and space," to appear, *Behavioral and Brain Sciences*, June 1985.

Feldman, J.A., "Dynamic connections in neural networks," *Biological Cybernetics*, *46*, 27-39, 1982.

Feldman, J.A. and D.H. Ballard, "Connectionist models and their properties," *Cognitive Science*, *6*, 205-254, 1982.

Feldman, J.A. and L. Shastri, "Evidential inference in activation networks," Proceedings, Cognitive Science Conf., Boulder, CO., June 1984.

Hinton, G.E., "A parallel computation that assigns canonical object-based frames of reference," *Proceedings*, 7th IJCAI, 683-685, Vancouver, B.C., August 1981a.

Hinton, G.E., "Shape representation in parallel systems," *Proceedings*, 7th IJCAI, 1088-1096, Vancouver, B.C., August 1981b.

Hinton, G.E. and J.A. Anderson (eds.), *Parallel Models of Associative Memory*. Hillsdale, NJ: L. Erlbaum Associates, 1981.

Hrechanyk, L.M. and D.H. Ballard, "Viewframes: A connectionist model of form perception," *Proceedings*, *CVPR*, Washington, D.C. June 1983.

Hummel, R.A. and S.W. Zucker, "On the foundations of relaxation labeling processes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, **PAMI-5**, 267-287, 1983.

Jolicoeur, P. and S.M. Kosslyn, "Coordinate systems in the long-term memory representation of three-dimensional shapes," *Cognitive Psychology*, **15**, 301-345, 1982.

Just, M.A. and P.A. Carpenter, "Eye fixations and cognitive processes," *Cognitive Psychology*, **8**, 410-480, 1976.

Kanade, T., "A Theory of Origami World," TR CMU-CS-78-144, Dept. of Computer Science, Carnegie Mellon Univ., 1978.

Marr, D.C., *Vision*. San Francisco, Ca: W.H. Freeman and Co., 1982.

Marr, D.C. and H.K. Nishihara, "Rrepresentation and recognition of the spatial organization of three-dimensional shapes," *Proceedings, Royal Society of London, Series B*, **200**, 269-294, 1978.

Minsky, M., "K-Lines: A theory of memory," *Cognitive Science*, **4**, 2, 117-133, 1980.

Minsky, M., "A framework for representing knowledge," In: *Psychology of Vision*, P. Winston (ed.), McGraw-Hill, 1975.

Minsky, M. and S. Papert. *Perceptrons*. Cambridge, MA: The MIT Press, 1972.

Palmer, S.E., E. Rosch, and P. Chase, "Canonical perspective and the perception of objects," In J. Long and A. Baddeley (eds.) *Attention and Performance IX*, Hillsdale, NJ: L. Erlbaum Associates, 1981.

Pentland, A.P., "Shading into texture," *Proceedings*, AAAI-84, 269-273, 1984.

Plaut, D.C., "Visual recognition of simple objects by a connection network," TR143, Computer Science Dept., Univ. of Rochester, August 1984.

Poggio, T. and V. Torre, "Ill-posed problems and regularization analysis in early vision,", *Proceedings*, DARPA Image Understanding Workshop, 257-263, October 1984.

Posner, M.I. *Chronometric Explorations of Mind*. L. Erlbaum Associates, 1978.

Posner, M.I. and Y. Cohen, "Components of visual oreinting," In: *Attention and Performance X*, H. Bouma & D. Bouwhis (eds.), L. Erlbaum Associates. 1984.

Sabbah, D., "Computing with connections in visual recognition of Origami objects," *Cognitive Science*, Special Issue on Connectionist Models, Winter 1985.

Sabbah, D., "Design of a highly parallel visual recognition system," *Proceedings*, 7th IJCAI, Vancouver, B.C., August 1981.

Shastri, L. and J.A. Feldman, "Semantic networks and neural nets," TR131, Computer Science Dept., Univ. of Rochester, June 1984.

Thompson, P., "Margaret Thatcher: A new illusion," *Perception*, **9**, 483-282, 1980.

Treisman, A.M., "The role of attention in object perception," *Proceedings*, The Royal Society International Symposium on Physical and Biological Processing of Images, London, September 1982.

Treisman, A.M. and G. Gelade, "A feature-integration theory of attention," *Cognitive Psychology*, *12*, 97-136, 1980.

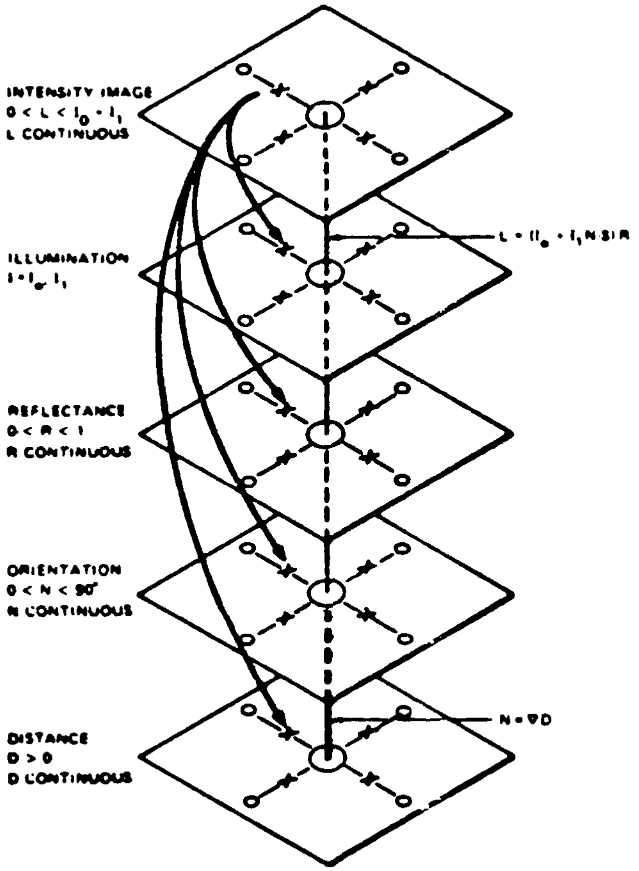


FIG 1.1: A PARALLEL COMPUTATIONAL MODEL FOR RECOVERING INTRINSIC IMAGES.

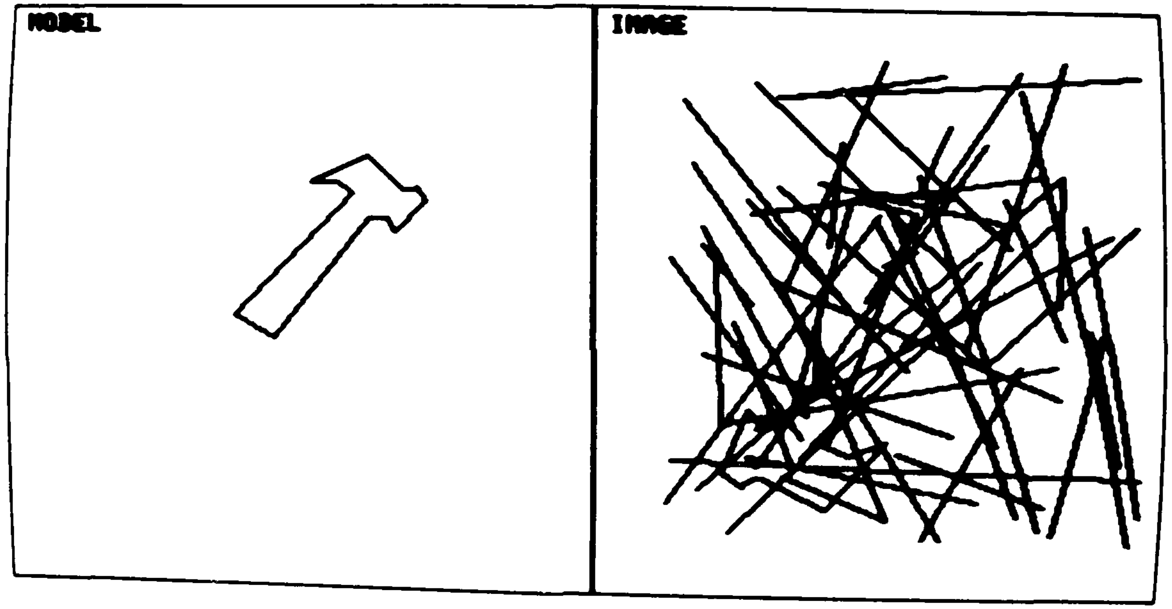


Figure 1.2: Recognizing a known object

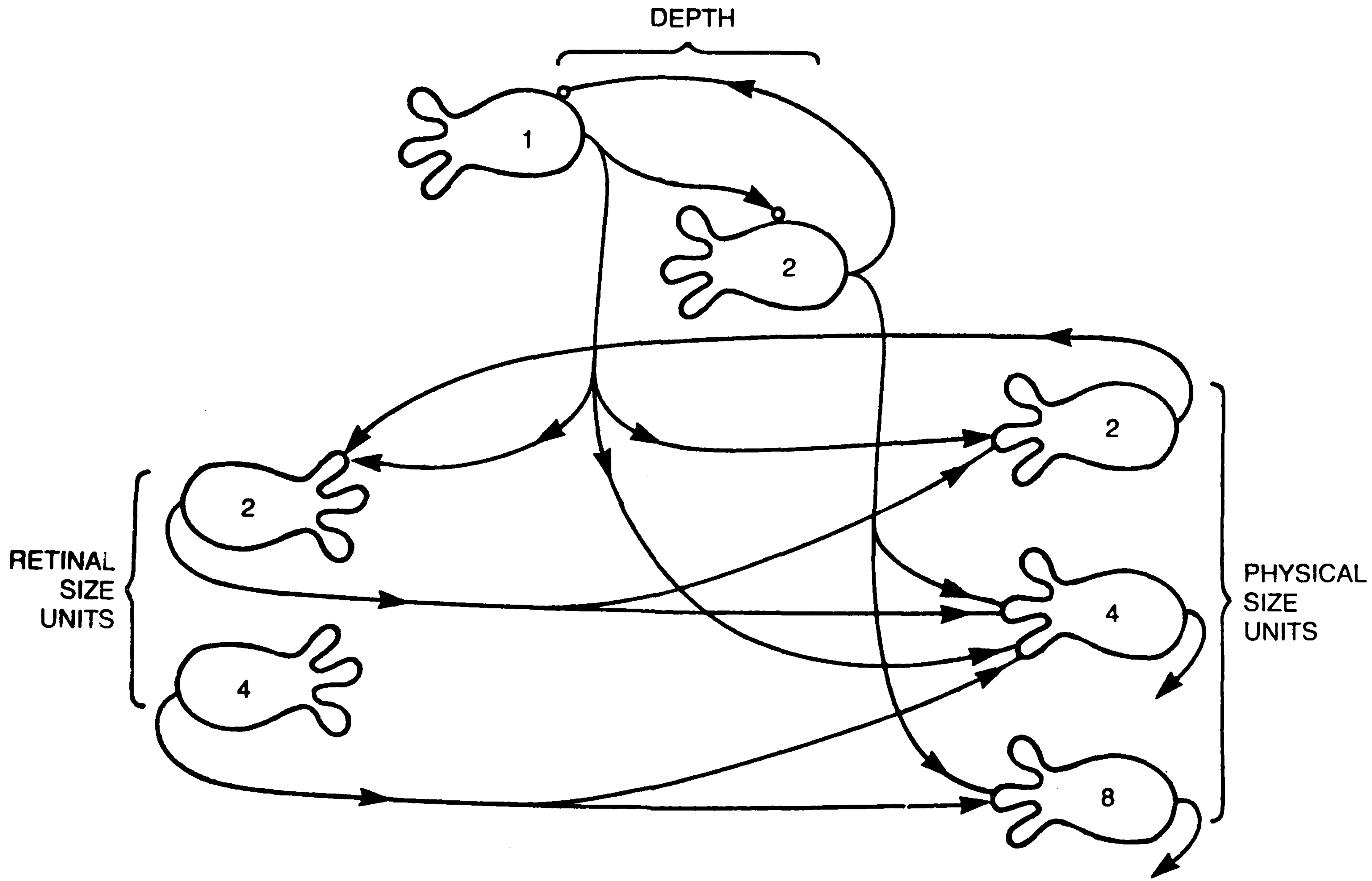


Figure 2.1: Relations among depth, retinotopic size, and physical size

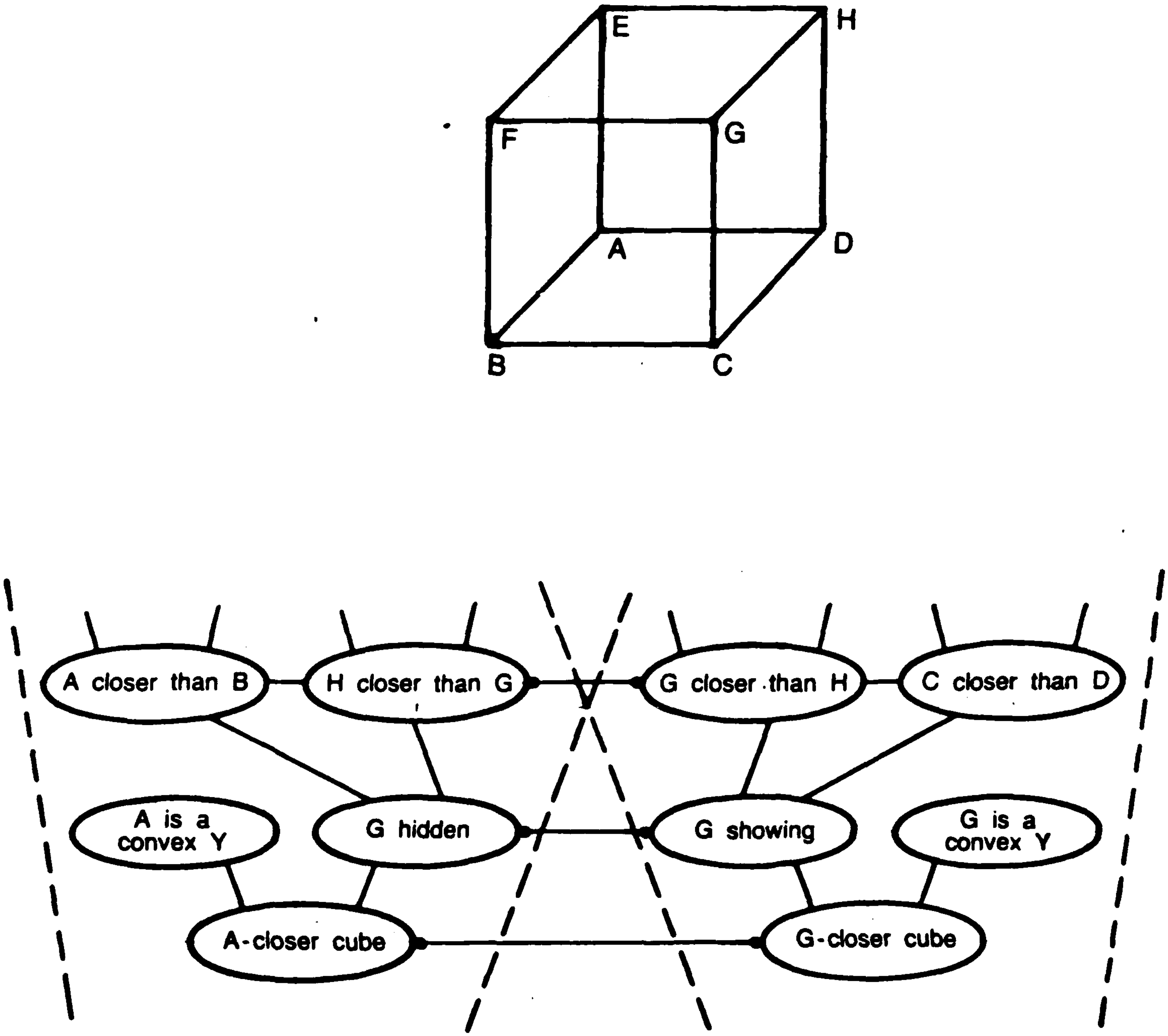
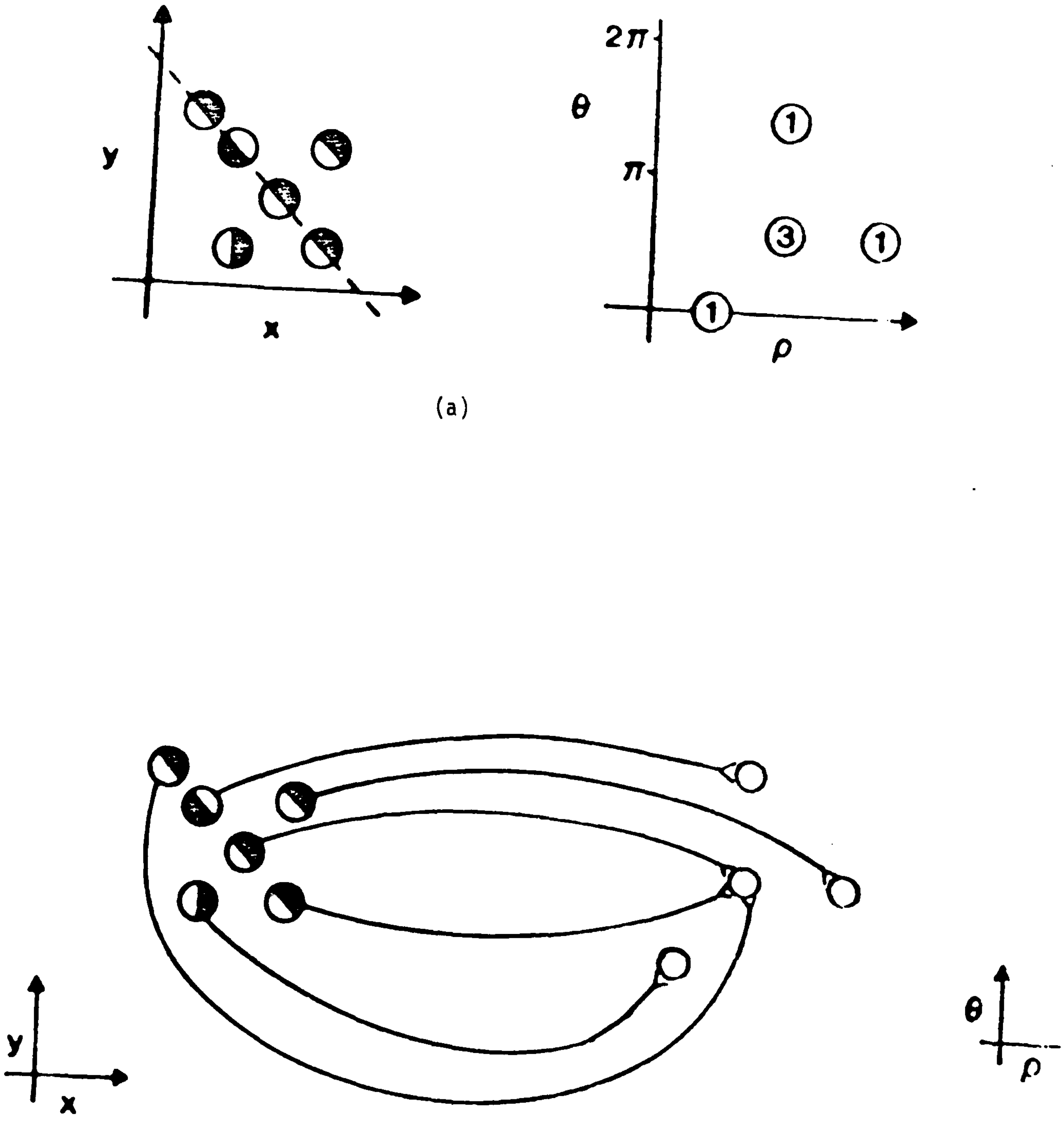


Figure 2.2: The Necker cube



(a)

(b)

Figure 2.3: Hough transform for lines

(a) Counting θ and ρ values suggested by edges

(b) Connectionist network for the calculation

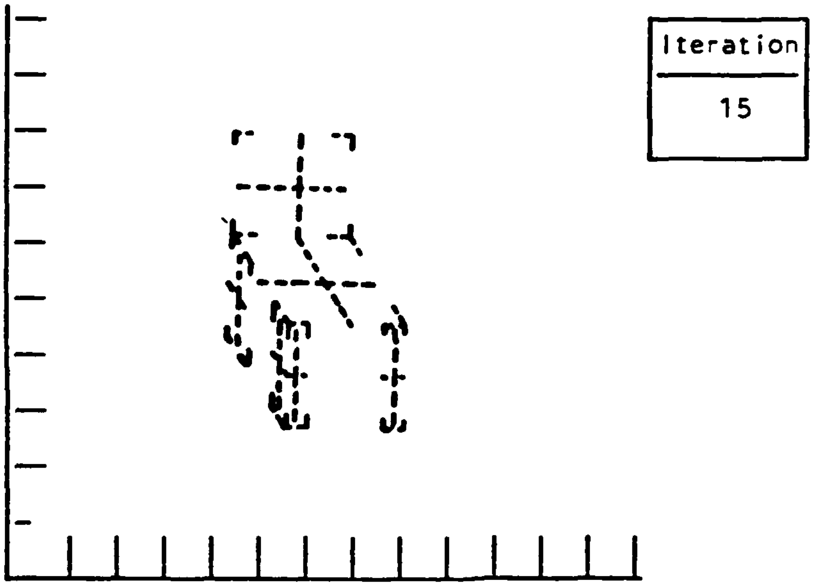


Figure 3.1: Origami Chair Behavior (3)

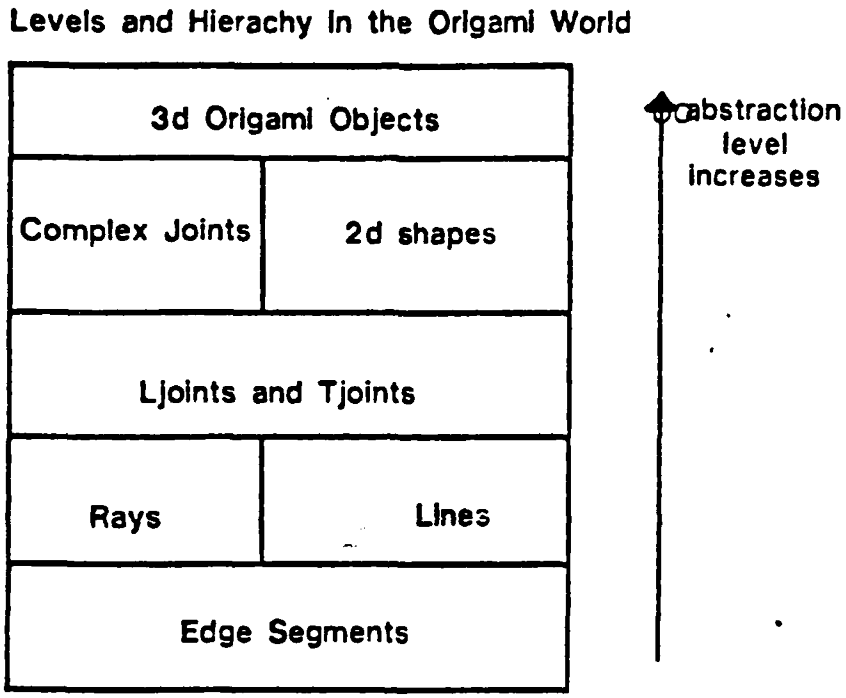


Figure 3.2: The Origami World Hierarchy

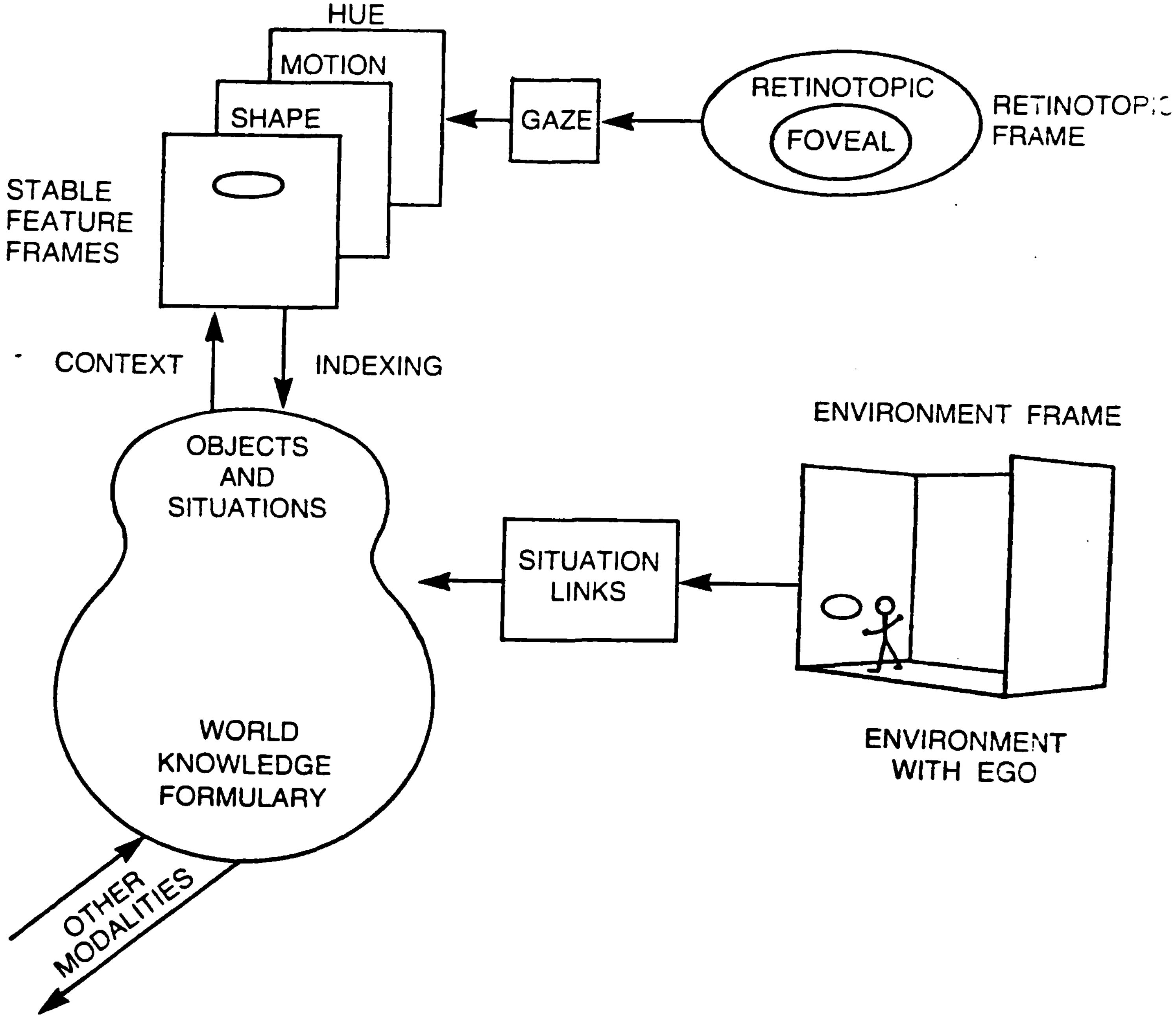


Figure 3.3: Four Frames, major structures and links

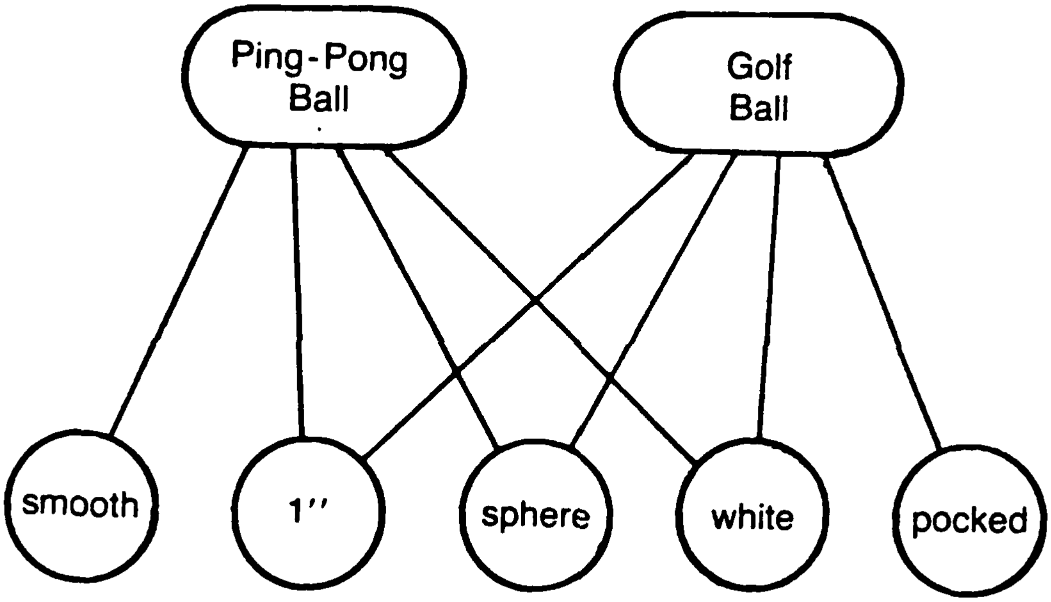


Figure 3.4: Ping-pong and golf balls

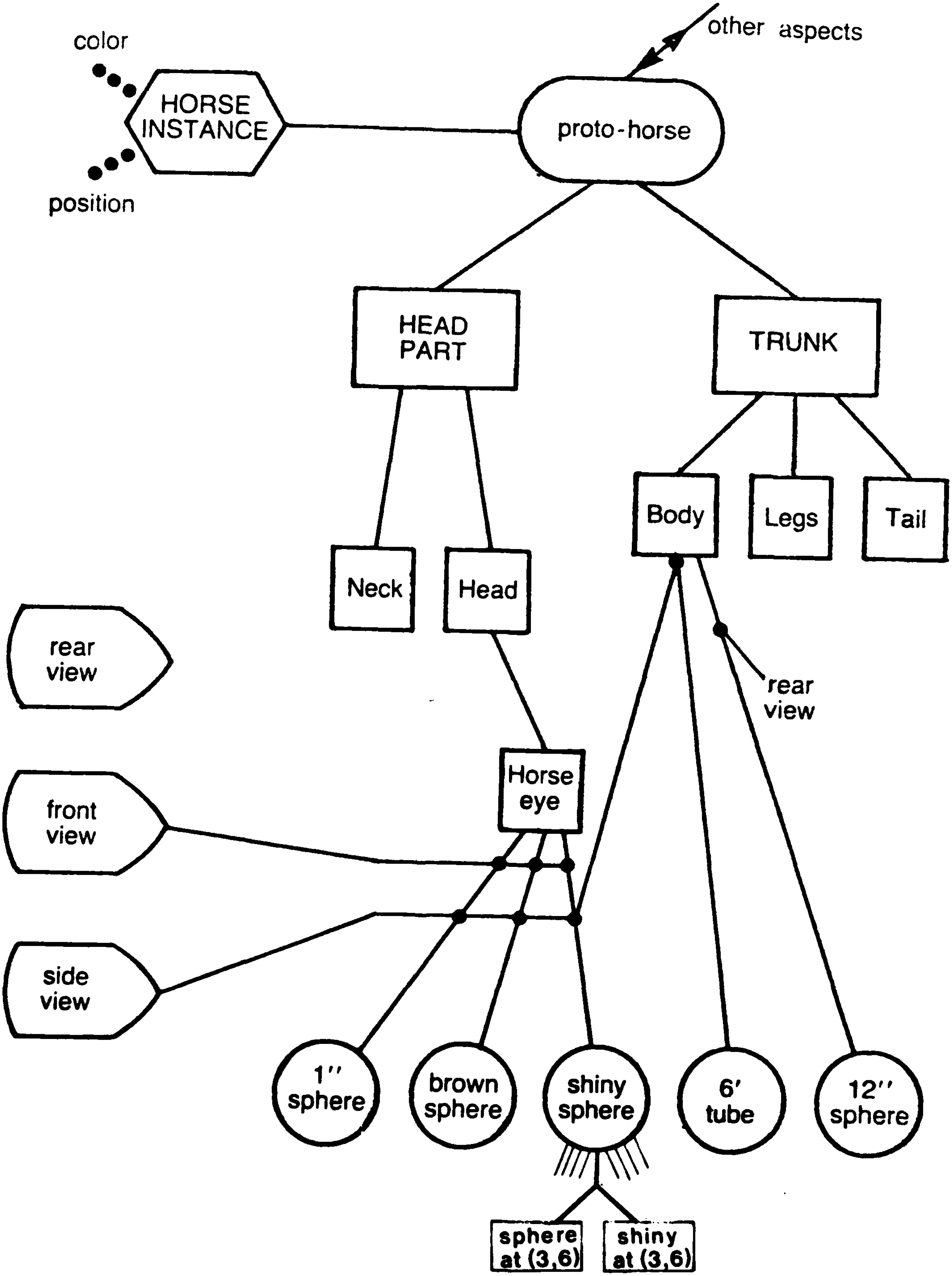


Figure 3.5: General view of horse

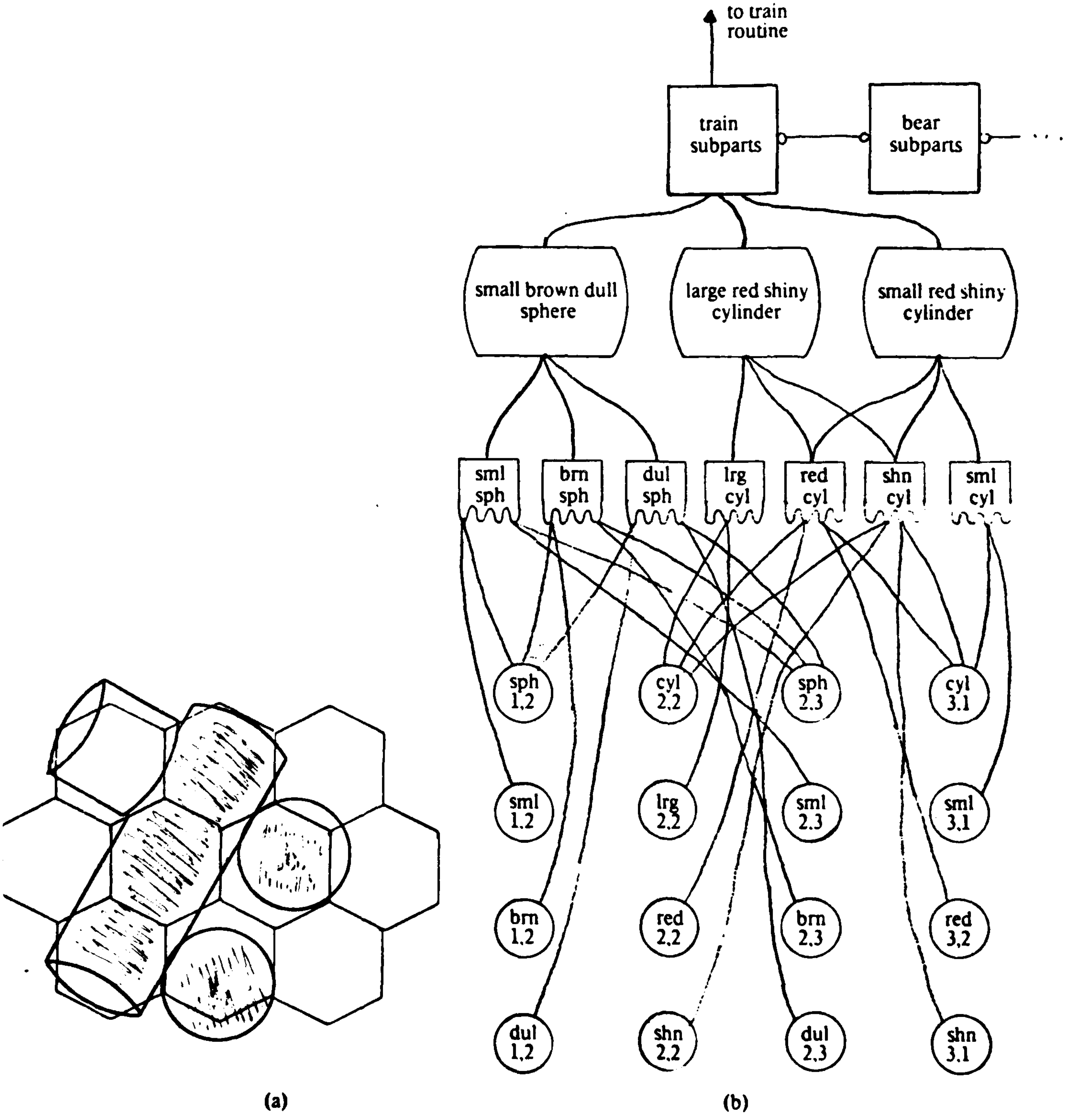


Figure 4.1:(a) possible feature input, (b) organization of portion of SIM for analyzing such input.

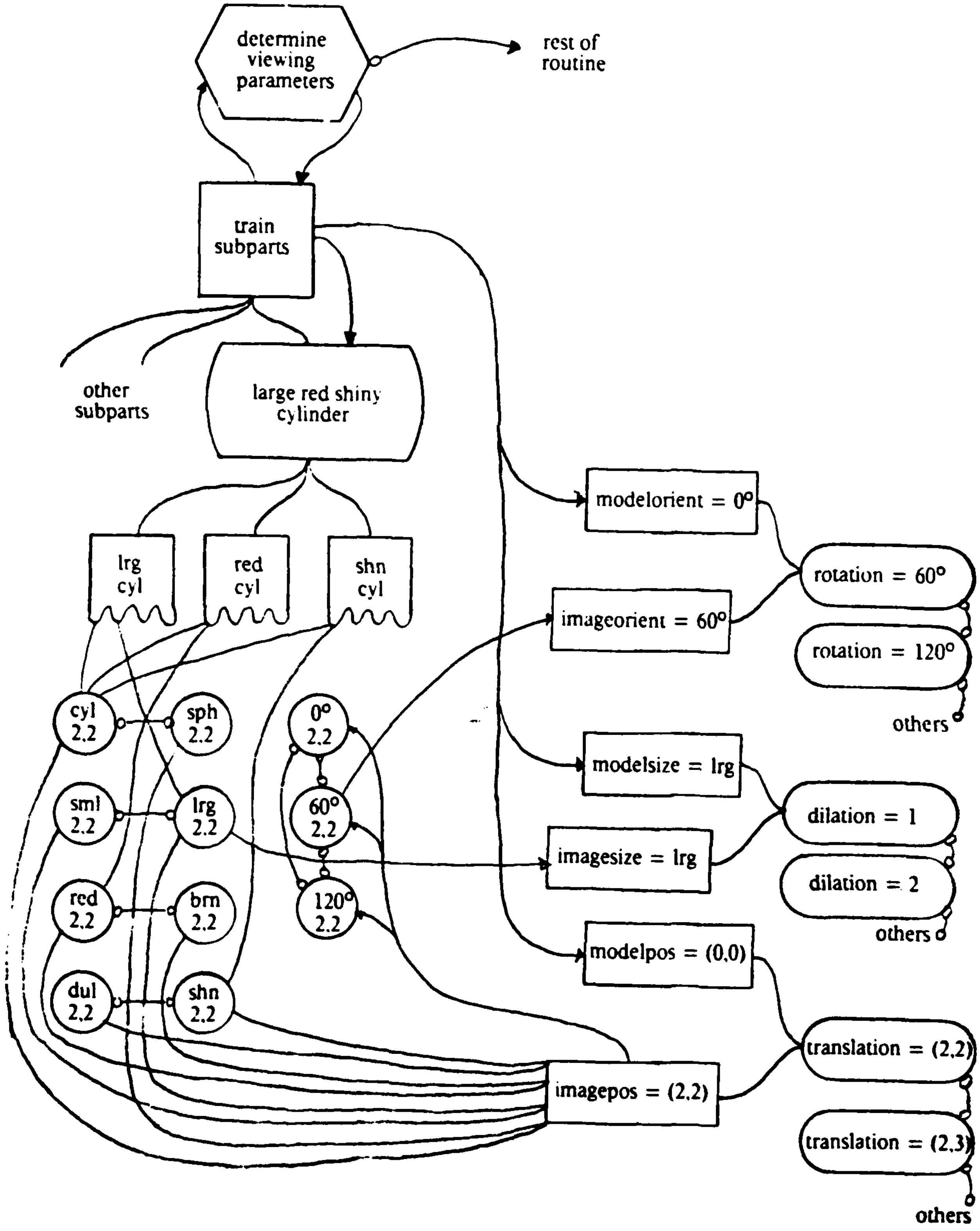


Figure 4.2: Determination of viewing transform parameters

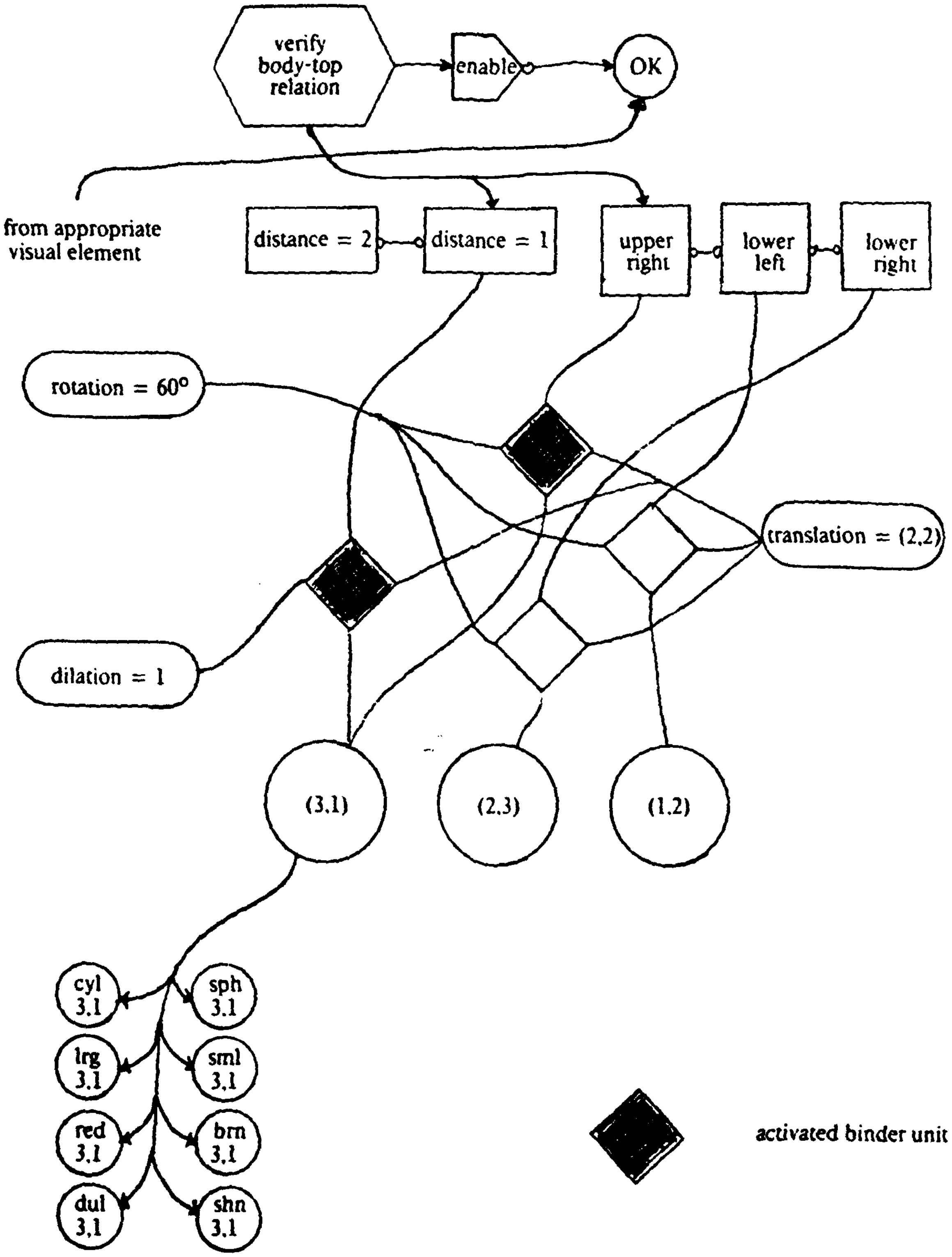


Figure 4.3: Structure of the VTM

END
DTIC
4-86